UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/844,673	04/30/2001	Radhika Thekkath	MTEC006/00US	8988

			EXAMINER
22903	7590	08/19/2005	RUTTEN, JAMES D

COOLEY GODWARD LLP
ATTN: PATENT GROUP
11951 FREEDOM DRIVE, SUITE 1700
ONE FREEDOM SQUARE- RESTON TOWN CENTER
RESTON, VA 20190-5061

ART UNIT	PAPER NUMBER
2192	

DATE MAILED: 08/19/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

	Application No.	Applicant(s)
Office Action Summary	09/844,673	THEKKATH, RADHIKA
	Examiner	Art Unit	
	J. Derek Rutten	2192	

-- *The MAILING DATE of this communication appears on the cover sheet with the correspondence address* --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed
  after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any
  earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on <u>03 June 2005</u>.

2a)☒ This action is **FINAL**.   2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is
closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-3,5,6,10,11,13,15-17 and 20-24* is/are pending in the application.

  4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-3, 5, 6, 10, 11, 13, 15-17, and 20-24* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

  Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

  Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

  a)☐ All b)☐ Some * c)☐ None of:

  1.☐ Certified copies of the priority documents have been received.

  2.☐ Certified copies of the priority documents have been received in Application No. _____.

  3.☐ Copies of the certified copies of the priority documents have been received in this National Stage
  application from the International Bureau (PCT Rule 17.2(a)).

  * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☒ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date <u>6/3/05</u>.

4)☐ Interview Summary (PTO-413)
   Paper No(s)/Mail Date. _____ .

5)☐ Notice of Informal Patent Application (PTO-152)

6)☐ Other: _____ .

DETAILED ACTION

1.   Acknowledgement is made of Applicants' amendment dated 03 June 2005 responding to the 04 January 2005 Office Action provided in the rejection of claims 1-3, 5, 6, 10, 11, 13, 15-17, and 20-24, wherein claims 1, 11, 21, 22, and 24 have been amended, and no claims were canceled. Claims 1-3, 5, 6, 10, 11, 13, 15-17, and 20-24 remain pending in the application and have been fully considered by the examiner.

2.   Applicant's arguments, see 2nd paragraph of page 10, filed 3 June 2005, with respect to the rejection of claim 1 under 35 U.S.C. 103(a) have been fully considered and are persuasive. Therefore, the rejection has been withdrawn. However, upon further consideration, a new ground(s) of rejection is made in view of U.S. Patent 5,870,606 to Lindsey.

3.   Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a). ·

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Response to Arguments

4.   Applicant's arguments on pages 9 and 10 of the response regarding new limitations, addressed in paragraph (2) above, are convincing.

5.   Applicant argues in paragraph 3 on page 10 of the response that the cited references do not teach tracing in response to a combination of a processor mode and an ASID value. In response to applicant's arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986). Further arguments in this paragraph regarding new limitations are addressed in the rejection of claim 1 below.

6.   Applicant's arguments in paragraph 4 on page 10 regarding claim 10 fail to comply with 37 CFR 1.111(b) because they amount to a general allegation that the claims define a patentable invention without specifically pointing out how the language of the claims patentably distinguishes them from the references.

Claim Rejections - 35 USC § 103

7.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

8.   Claims 1, 3, 5, 6, 10, 11, 13, 15-17, and 20-24 are rejected under 35 U.S.C. 103(a) as being unpatentable over prior art of record U.S. Patent 5,621,886 to Alpert et al. (hereinafter

"Alpert") in view of prior art of record "MIPS64 5Kc™ Processor Core Datasheet" by MIPS

Technologies (hereinafter "5Kc"), further in view of U.S. Patent 5,870,606 to Lindsey

(hereinafter "Lindsey").

In regard to claim 1, Alpert discloses:

A tracing method (See column 16 line 12 – column 17 line 25), *comprising:*

detecting a processor mode of a processor See column 4 lines 23-24

This first indication indicates which mode the processor is currently operating in.

and <an address range> *defining an identity of a task being run on said processor;* See

column 7 lines 7-10:

> Each address breakpoint register is used for storing a breakpoint address, while each of
> the breakpoint mask registers are used for storing a mask. Each breakpoint address and its
> corresponding breakpoint mask define an address range.

receiving control signals defining <range> *values and processor modes for which*

tracing is triggered; See column 7 lines 14-18

> In response to receiving designated addresses from internal bus 144, this circuitry
> determines whether these addresses are within any of the address ranges defined by the
> information in the address breakpoint registers and the breakpoint mask registers.

This passage is exemplary for receiving address control signals. However, control

signals for processor modes are inherent since CPUs inherently use control signals to

define the operation of the processor, and would be inoperable without them.

effecting a predefined tracing control based on a logical comparison of a current

processor mode and a current <range> *value to said control signals, whereby tracing is*

triggered for selected processor modes and <range> *values.* See column 4 lines 31-36:

> The debug circuitry is also coupled to the circuit to receive either the second indication or
> the third indication based on the state of the first indication. The **debug circuitry allows**

for the recognition of the debug event based on the state of the indication it receives
from the circuit.

storing trace information in a trace memory and See column 8 lines 24-29:

> In the described embodiment, processor 110 **stores the current process' execution
> environment** by copying the contents of instruction pointer register 147 and status
> register 150 **into suspended instruction pointer register 148 and suspended status
> register 160**, respectively.

applying trace regeneration software to said trace information to reconstruct said

trace information after said task is executed on said processor. See column 1 lines 36-48

in support of column 8 lines 49-52:

> Upon recognizing an event, a processor's event handling mechanism causes the processor
> to interrupt execution of the current process, **store the interrupted process'execution
> environment** (i.e., the information necessary to resume execution of the interrupted
> process), and **invokes the appropriate operating system handler.** Upon completing the
> servicing of the event, the invoked handler instructs the processor to resume execution of
> the interrupted process using the previously stored execution environment. The
> appropriate handler for debug events is the debug handler. Thus, in response to each
> debug event, the debug handler is executed. **The debug handler is a portion of the
> debug program and it allows the programmer to perform a variety of debugging
> techniques.**
> ...
> As shown in step 230, the status register is loaded with the appropriate values for the
> execution of the appropriate handler and flow passes to step 240.

Note that reconstruction of trace information is inherently performed by the debug

handler by reading the values of the instruction pointer register and suspended status

register. Without reading these trace memories, a debug handler would be unable to

reconstruct the trace information necessary for performing the variety of debugging

techniques.

While Alpert discloses limiting debug events for specific address ranges, it does

not expressly teach a singular ASID value for enabling debug events. Also, Alpert does

not expressly disclose the reconstruction of trace information *after* execution.

However, in an analogous environment, 5Kc teaches that hardware breakpoints

can be triggered using ASID values. See page 15 column 2, "Hardware Breakpoints":

> Hardware breakpoints are provided as an optional feature. Four instruction breakpoints and two data breakpoints are supported. Depending on how the debug resources are programmed, a debug exception is taken when a hardware breakpoint matches, whereby **the normal application is suspended and debug mode is entered**.
> Debug instruction breaks occur on executed instructions also when executed from the cache. Instruction breaks are set on the instruction virtual address and can also **compare the ASID value** used by the MMU. Finally, a bit mask can be applied to the virtual address to set breakpoints on a range of instructions.
> Debug data breakpoints occur on explicit load/store accesses. **Breakpoints are set on the virtual address and ASID value**, similar to the instruction breakpoint. Data breakpoints can be set on a load and/or store access. Data breakpoints can also be set based on the data value of the load/store operation. Finally, masks can be applied to both the virtual address and the load/store data value.

Also in an analogous environment, Lindsey teaches that trace information can be

utilized after execution of a task. See column 4 lines 45-49:

> When the trace function is triggered, **the debugger collects trace information** regarding what is occurring **while the program being debugged is executing. This information is then made available to the developer** for use in attempting to debug the program.

It would have been obvious to one of ordinary skill in the art at the time the

invention was made to use 5Kc's ASID values with Alpert's debug circuitry. One of

ordinary skill would have been motivated to set a breakpoint on a specific application that

might have multiple address representations. Also, it would have been obvious to one of

ordinary skill in the art at the time the invention was made to use Lindsey's teaching of

trace utilization with Alpert's trace information. One of ordinary skill would have been

motivated to capture information relating to a sequence of memory accesses including

data movement relative to various data components (see Lindsey column 1 lines 40-65).

As per claim 3, the above rejection of claim 1 is incorporated. Alpert further discloses: *wherein an indication of said control input for said current operating state is obtained via a software-settable trace control register* (column 8 lines 49-51).

As per claim 5, the above rejection of claim 1 is incorporated. Alpert further discloses: *wherein said processor modes comprise at least one of a kernel mode, a supervisor mode, a user mode, and a debug mode* (column 6 lines 1-3 provides the at least one kernel mode).

In regard to claim 6, the above rejection of claim 5 is incorporated. Alpert does not expressly disclose a MIPS32™ or MIPS64™ architecture specification. However, 5Kc teaches the implementation of a processor based on the MIPS64™ architecture (page 1), and compatibility of this processor with those based on the MIPS32™ architecture (page 7).

As per claim 10, the above rejection of claim 1 is incorporated. Alpert further discloses: *wherein said tracing is triggered based on <address>, U, and K, controls, wherein trace data is processed for a current <address> value, U, if asserted, enables tracing in user mode, K, if asserted enables tracing in kernel mode, said controls enabling tracing when:*

   ((<address> equals a current process application space) AND

   ((U is asserted AND said processor is in user mode) OR

(K is asserted AND said processor is in kernel mode))) See column

4 lines 31-36 as cited in the above rejection of claim 1. Alpert provides debug support

when a processor is in a particular operating mode with a particular address range.

Control signals are inherent in the operation of a processor otherwise it could not

function.

Alpert does not expressly disclose *ASID, G, S, DM, and X controls, wherein said*

G if asserted implies that all processes are to be traced, whereas if G is not asserted,

trace data is processed for a current ASID value, S, if asserted, enables tracing in

supervisor mode, DM, if asserted, enable tracing in a debug mode, and X, if asserted,

enables tracing for exception and error level conditions, said controls enabling tracing

when:

    ((G is asserted) AND

        ((S is asserted AND said processor is in supervisor mode) OR

        (DM is asserted AND said processor is in debug mode) OR

        (X is asserted AND (an exception level bit is asserted OR an error

        level bit is asserted)))).

However, 5Kc teaches ASID control values (as cited in the above rejection of

claim 1), as well as user, supervisor, kernel, and debug modes (page 4 column 1

paragraph 1).

It would have been obvious to one of ordinary skill in the art at the time the

invention was made to use the teaching of 5Kc to enable the use of additional modes of

operation within Alpert's processor. Alpert provides the ability to control debug facilities

based on two processor modes. One of ordinary skill would have been motivated to enable fine-grained control of an application or system process using multiple processor modes and debug conditions.

As per claim 11, Alpert discloses a processor core and trace generation logic (Figure 1; also column 5 lines 44-50). All further limitations have been addressed in the above rejection of claim 1.

In regard to claim 13, 15, 16, and 20, the above rejection of claim 11 is incorporated. All further limitations have been addressed in the above rejection of claims 3, 5, 6, and 10, respectively.

As per claim 17, the above rejection of claim 1 is incorporated. Alpert further discloses: *wherein said at least one operating state includes an identity of a process being run on said processor and said predefined trace control is based on a current processor mode and said identity of a process* (column 5 lines 55-57; column 6 lines 1-3).

As per claim 21, Alpert discloses computer-readable program code (Figure 1, element 122). All further limitations have been addressed in the above rejection of claim 1.

As per claim 22, Alpert discloses transmitting code to a computer, as this is inherent in execution debug software, since a computer needs code in order to execute (Figure 1, element 122). All further limitations have been addressed in the above rejection of claim 1.

As per claim 23, the above rejection of claim 22 is incorporated. Alpert does not expressly disclose transmitting via the Internet. However, official notice is taken, since the Internet is a well-known medium for exchanging data between computer systems in different physical locations.

As per claim 24, Alpert inherently discloses a data signal embodied in a transmission medium (Figure 1, element 140). All further limitations have been addressed in the above rejection of claim 1.
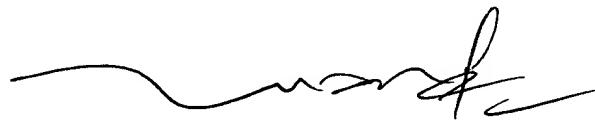
Conclusion

9.   Any inquiry concerning this communication or earlier communications from the examiner should be directed to J. Derek Rutten whose telephone number is (571) 272-3703. The examiner can normally be reached on T-F 6:00 - 4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent

Application Information Retrieval (PAIR) system. Status information for published applications

may be obtained from either Private PAIR or Public PAIR. Status information for unpublished

applications is available through Private PAIR only. For more information about the PAIR

system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR

system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

jdr

TUAN DAM
SUPERVISORY PATENT EXAMINER